Milo Piazza

mpiazza@berkeley.edu • 323-327-2602 • www.linkedin.com/in/milopiazza • github.com/Milo-Piazza

EDUCATION

University of California, Berkeley

- BA in Computer Science, Expected May 2019
- Computer Science GPA: 3.83

Relevant Coursework:

- Structure and Interpretation of Computer Programs, Machine Structures, Algorithms and Intractable Problems, Data Structures, Introduction to Artificial Intelligence, Database Systems, Computer Security

SKILLS

- **Programming languages:** C, C++, Objective-C, Python, Java, PHP, SQL, Haskell, Scheme
- **Libraries:** NumPy, Apache Spark, OpenMP
- **Software:** Microsoft Office, Adobe Photoshop, GIMP
- **Other Skills:** Creativity, Problem Solving, Martial Arts (13 years experience)

EXPERIENCE

Santa Monica College, Santa Monica, CA — Jul – Aug 2017

Computer Science Tutor

- Provided one-on-one assistance to summer students in C, C++, Python, and Java courses

UC Berkeley, Berkeley, CA — Aug – Dec 2016

Lab Assistant *(Computer Science 61A)*

- Aided students with Python programming assignments in lab sections and office hours

Bridges Academy, Studio City, CA — Aug 2014 – May 2015

Teaching Assistant

- Taught Python to high school students in Beginning Programming class

COMPETITIONS

FIRST Robotics Competition

Lead Programmer

- Oversaw Bridges Academy's FRC programming team to design and implement C++ code for two different robots
- Implemented automated instructions, user-controlled instructions, and a debug mode for each robot
- Mentored novice team programmers in robotics libraries

PROJECTS

Pacman AI

- Implemented various AI and machine learning algorithms in Python to allow the computer to autonomously play Pacman

Text Editor

- Developed in Java, using JavaFX, a fully functional text editor that can insert or remove from any text file in constant time

Wizards

- Designed and implemented a randomized local search algorithm to solve a specific type of constraint satisfaction problem involving finding an ordering of items
- Created a data structure that can store an ordering of items, move an item to a new position, and check if a constraint is violated all in constant time
- Devised optimizations that resulted in a substantial improvement in asymptotic runtime compared to a naïve implementation

BearMaps

- Wrote the back end for a basic web mapping client using Java
- Incorporated the OpenStreetMap XML format and JSON queries

Scheme Interpreter

- Created a complete interpreter for the Lisp dialect Scheme using Python